Tandy Vendor Development Internal Document
Number: VDS-263901-01

Microsoft Works Utility Disk

August 7, 1986

Dear Tandy 600 Software Developer:

Here is the Microsoft Works Utility Disk and documentation you have requested. Microsoft Work affords the software developer the ability to develop, maintain, and debug individual assembly language programs or alternate ROMs from an MS-DOS based computer.

In order to use Microsoft Works, the following is required:

- Microsoft Macro-Assembler (MASM) version 4.0.
- MS-DOS based computer with 512K RAM (not compatible with Tandy 2000).

Neither Tandy Corporation nor Microsoft will offer any support for Microsoft Works.

Tandy Corporation provides vendor-level technical support to those vendors involved in developing products for the Tandy line. If your company is not a member already, please write to the following for more information:

Vendor Development Services
1300 One Tandy Center
Fort Worth, TX 76102

Enclosures

Microsoft® Works™
Debugging Application Programs
January 3, 1986

Debugging Works™ programs requires an IBM PC or compatible with 512KB of memory. The Works™ environment is run on top of a BIOS simulator which emulates most of the I/O capabilities of a battery-powered portable computer.

The language and debugging utilities (which are part of the Microsoft® Macro Assembler package) required for developing and debugging Works™ applications are:

- -An assembler (MASM) or compiler that is compatible with the Microsoft® Linker.
- -Microsoft® Linker (LINK).
- -Microsoft® Symbolic Debug Utility (SYMDEB).
- -The MAPSYM utility which translates .MAP files (output by LINK) into .SYM files (read by SYMDEB).

The Works™ Development Utilities package includes:

- -A utility (EXECNV) that converts .EXE files to Works™ executables.
- -Microsoft® Works™ Rom Generation Program (BLDROM) that builds a ROM image from a specified list of MS-DOS files.
- -A sample .GEN file (DEBUG.GEN) which contains a list of files that should be built into debugging ROM images. DEBUG.GEN is passed as a command line argument to BLDROM.
- -Works™ system files (they have a .SYS extension) that must be built into debugging ROM images. Note that the first file in a debugging ROM image must be the operating system ($$SYS001.SYS).
- -Two application files, WORD and CALC.
- -A batch file (MAKEDBG.BAT) that executes BLDROM and deletes any old RAM image files.
- -An MS-DOS-based BIOS simulator (HHSIM) and associated parameter file (DEBUG.PRM) on top of which a Works™ ROM image can be executed.
- -Symbol table files (HHSIM.SYM, HHOS.SYM) containing the symbols necessary for stepping through the simulator and operating system in order to reach the entry point of an application.
- -A batch file (HHDEBUG.BAT) that brings up HHSIM under SYMDEB.

The following steps should be followed in order to debug a Works™ application after a ROM image has been built:

Enter the following DOS command:

HHDEBUG app_symbol_table_filename

Enter the following SYMDEB commands:

G START
D ROMADR L2
SYMDEB output will be of the form, 'seg:off xx yy'.
XO HHOS!
Z OSCODE yyxx
G OSCODE:EXEC
T
XO HHOS!
G OSCODE:EXEC
After the System Manager comes up, RUN the application that is to be debugged.
T
XO app_symbol_table_filename!

Proceed with debugging the application.

Microsoft® Works™
BLDROM and EXECNV
Development Utilities
January 3, 1986

Command Line Arguments

The BLDROM.EXE utility will construct a ROM image that can be made part of a Works™ ROM file system. The utility takes two command line arguments as follows:

BLDROM <GENERATION FILENAME> <OUTPUT FILENAME>

The generation file, explained in more detail below, contains a list of the MS-DOS™ files that the ROM image should be constructed from. The name of the generation file must be fully specified; there is no default extension. The output filename is an optional argument. It can be used to specify the file that the resultant ROM image should be written to. The ROM image will be written to "HHC.ROM" if no output file is specified. If a filename without an extension is specified, then "ROM" will be used as the default extension.

Generation File

In the generation file, there must be one line per file that is to be put in the ROM image. A line in the generation file must have the following format:

<MS-DOS™ FILENAME> <WORKS™ FILENAME>

The MS-DOS™ file is the one containing the bytes which have to be put in a ROM file. Note that it doesn't matter what the semantics of the bytes are - the file could contain either code for an application or arbitrary data. The Works filename indicates what name the file within the ROM image should have. Both filenames must be specified in their entirety; there are no default extensions.

ROM Applications

Once an application has been written and *structured* as described in the *Microsoft® Works™ Operating System Programmers' Reference Guide*, it should be assembled or compiled into an MS-DOS™ ".EXE" file. This file can appear as the MS-DOS™ filename on a line in the generation file. BLDROM will automatically strip off the ".EXE" header to create a Works executable file. The Works executable file will be named according to the second filename on the line in the generation file.

RAM Applications: Use EXECNV

In the event that an application is to be stored on disk and run in RAM, the EXECNV utility should be used by itself (BLDROM doesn't get used at all) to perform the ".EXE" conversion. EXECNV takes a ".EXE" file as its only argument (leave out the ".EXE" because it is assumed) and produces a ".HHX" file. The ".HHX" file is ready to be run in a Works environment.

ROM Image Structure

A ROM image containing n files will have the following structure:

32 byte ROM image header
32 byte file header
ROM file #1

.
.
.

32 byte file header
ROM file #n
32 byte ROM image terminator

Remember to allow for header and terminator overhead when trying to anticipate whether or not a ROM image will fit in a ROM memory region.

Microsoft[R] Works[TM]
HHSIM
Development Utility
January 3, 1986

The HHSIM utility simulates a Works[TM] environment under MS-DOS[TM] on an IBM PC. The utility emulates a Works[TM] BIOS as if the IBM PC were a battery-powered portable computer.

The simulator is invoked with the MS-DOS[TM] command 'HHSIM parameter_filename'. If a parameter file is not specified, then HHC.PRM is assumed. If HHC.PRM is not found, then built-in defaults will be used.

In the parameter file there must be exactly one parameter per line; blank lines are not allowed. The following are the currently recognized parameters along with their default values:

RAMSIZ = 128	: size of RAM in K bytes.
RAMFILL = hex_byte_value	: default is not to fill.
TXTLIN = 25	: no. of display lines to use.
TXTCOL = 80	: no. of display columns to use.
SAVE = filename	: default is not to save.
ROM = HHC.ROM	: name of ROM image file.

The RAMSIZ parameter specifies the amount of PC memory that should be used to emulate portable computer RAM. However, there is no corresponding ROMSIZ parameter because the simulator automatically allocates enough PC memory for the ROM image. The RAMFILL parameter specifies a hexadecimal byte value with which to initially fill the area of PC memory that is used to emulate portable computer RAM. The SAVE parameter specifies the name of a RAM image file to either load when starting or produce when terminating if <ALT-F10> was pressed.

The simulation can be terminated by pressing either <ALT-F9> or <ALT-F10>. <ALT-F10> has the added effect of saving the RAM image if a SAVE parameter was specified in the parameter file.

When running the simulation, the PC's peripherals are accessed as if they were connected to a portable, except for hard disks, printers and asynchronous serial communication hardware. The simulator only supports floppy disks and does not provide support for printers and communications.